

ARGUMENTS/ REMARKS

Favorable reconsideration of this application, as presently amended and in view of the following discussion, is respectfully requested.

Claims 1-17 are pending in the present application. Claims 1, 2, 10 and 11 are amended by the present amendment.

Claim amendments find support in the specification as originally filed at least at page 1, line 12, and page 14, lines 14-25. Thus, no new matter is added.

In the outstanding Office Action, Claims 1-17 were rejected under 35 U.S.C. § 112, second paragraph; Claims 1, 2 and 4-17 were rejected under 35 U.S.C. § 103(a) as unpatentable over U.S. Patent No. 6,269,409 to Solomon in view of U.S. Patent No. 5,623,677 to Townsley et al. (herein "Townsley"); and Claim 3 was indicated as allowable if rewritten in independent form.

Initially, Applicant gratefully acknowledges the indication of allowable subject matter in Claim 3.

Further, regarding the rejection of Claims 1-17 under 35 U.S.C. § 112, second paragraph, Claims 1, 2, 10 and 11 are amended to more clearly indicate that OSs refer to Operating Systems, and Claims 1 and 11 are amended to indicate that the primary OS determines whether there exists any task to be executed on the secondary OS. In addition, Applicant respectfully traverses the assertion in the outstanding Office Action that "it is unclear what no task means" in Claims 2 and 10. Applicant respectfully submits that one of skill in the art of OS design would know that a function of an operating system is to schedule tasks for execution by a processor and that a task exists to be executed on an OS when the task is scheduled for execution by the OS, for example as described in Applicant's specification at page 14, lines 14-25. Accordingly, Applicant respectfully requests the rejections of Claims 1-17 under 35 U.S.C. § 112, second paragraph, be withdrawn.

In addition, Applicant respectfully traverses the rejection of Claims 1, 2 and 4-17 under 35 U.S.C. § 103(a) as unpatentable over Solomon in view of Townsley.

Claim 1 is directed to a processor power-saving control method that includes, in part, a step by a primary OS of determining whether there exists any task to be executed on a secondary OS. When the primary OS determines that there is a task to be executed on the secondary OS, the primary OS interrupts the secondary OS by issuing an interrupt to the secondary OS. Claims 2, 10 and 11 include similar features. This approach may advantageously reduce device power consumption.¹

Applicant respectfully submits that the disclosures of Solomon and Townsley, whether taken individually or in combination, do not teach or suggest each of the features of the independent claims. For example, Solomon and Townsley do not teach or suggest any power-saving control method that includes issuing an interrupt to a secondary OS when a primary OS determines there is a task to be executed on a secondary OS. Solomon describes a software abstraction layer that allows a first OS to run concurrently with a second OS.² In the method of Solomon, a UNIX operating system (e.g., a first OS) may determine that an interrupt 504 generated by a base machine 502 is “for a device within Windows NT operating system 508”³ (e.g., a second OS), and if so, may report the occurrence of the interrupt to a software abstraction layer 512. Subsequently, the Windows NT operating system may call the software abstraction layer 512 in response to the reported interrupts.⁴ However, Solomon is silent regarding whether or not the Windows NT operating system or the UNIX operating system (e.g., first OS) consider if there are any tasks to be executed by the Windows NT operating system (e.g., second OS), and Solomon does not indicate that the UNIX operating system interrupts the Windows NT operating system when there are tasks to be executed by

¹ Specification at page 18, lines 1-3.

² Solomon at Abstract.

³ Solomon at column 4, lines 32-42.

⁴ Solomon at column 4, lines 48-52.

the Windows NT operating system. Further, Applicants respectfully note that Solomon includes no motivation for the UNIX operating system to determine if there are tasks to be executed in the Windows NT operating system.

Accordingly, Applicant respectfully submits that Solomon does not teach or suggest “when the primary OS determines there exists any task to be executed on said secondary OS, interrupting said secondary OS by issuing a secondary-OS interrupt from the primary OS to the secondary OS,” as recited in independent Claim 1, and as similarly recited in independent Claims 2, 10 and 11.

Further, Townsley merely describes disabling a clock of a processor to place the processor in an inactive state and thereby reduce a power consumption of the processor.⁵ However, Townsley is silent regarding the claimed features that are lacking in the disclosure of Solomon. For example, Townsley is silent regarding any first OS that determines if there exist any task to be executed on a second OS, and Townsley is likewise silent regarding interrupting any second OS from a first OS in any situation. In addition, Applicants respectfully submit there is no suggestion or motivation in Townsley to achieve the claimed invention by combination with Solomon.

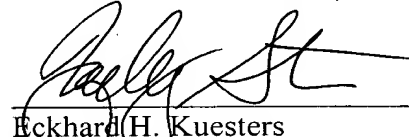
Accordingly, Applicant respectfully submits that independent Claims 1, 2, 10 and 11, and claims depending therefrom, are allowable.

⁵ Townsley at column 17, lines 62-67.

In view of the foregoing discussion, no further issues are believed to be outstanding in the present application. Therefore, Applicant respectfully requests that this application be allowed and be passed to issue.

Respectfully submitted,

OBLON, SPIVAK, McCLELLAND,
MAIER & NEUSTADT, P.C.



Eckhard H. Kuesters
Registration No. 28,870

Customer Number

22850

Tel: (703) 413-3000
Fax: (703) 413 -2220
(OSMMN 06/04)

Zachary S. Stern
Registration No. 54,719

EHK:ZSS:dnf